

EL SENCILLO TUTORIAL DE MOTIF DE HAYLEM CANDELARIO BAUZá

Yo. Haylem Candelario Bauzá:

Me he decidido crear este sencillo manual de Motif, debido a la falta de un documento actualizado del mismo. Los manuales que he revisado contienen código obsoleto con una sintaxis antigua que rara vez compila con compiladores modernos. Por esa misma razón he tratado de recopilar ejemplos que resuelven tareas cotidianas y de interés para así a mí mismo srvirme de referencia a futuros proyectos. Por otro lado Motif no es malo!!! ni mucho menos esta obsoleto como muchos piensan; producto al bombo y platillo ue se le da a lo nuevo y lo mas popular, quedan en el pasado las reales buenas ideas. Cuando realmente se escribía softwre bien pensado y útil y al que quiera continuar creando aplicaciones potentes, rápidas y serias, aquí les dejo mi granito de arena para que así esta maravillosa librería pueda ser entendida y valorada. Las interfaces modernas carecen de este encanto y poseen muchos errores visuales como saltos inesperados en la interfaz, ralentiaciones y sobrecarga visual innecesaria. Por otro lado, la elegancia, seriedad y confiana que generan este tipo de aplicaciones legacy es impagable. Larga vida a Motif!!!!.

Sobre la librería motif:

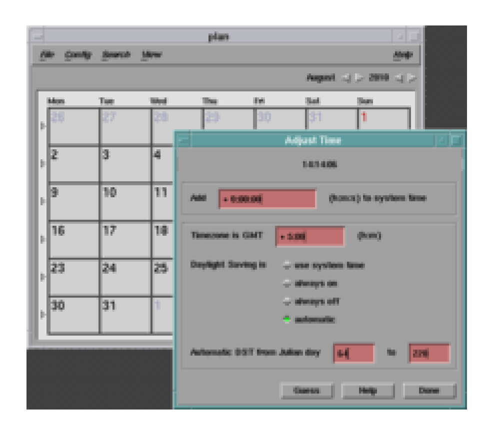
Motif es una biblioteca para la creación de entornos gráficos bajo X Window System en sistemas Unix. Motif es también un estándar de la industria bajo el código IEEE 1295. Actualmente propiedad de The Open Group. Su última versión es la 2.3.6 de junio de 2016.

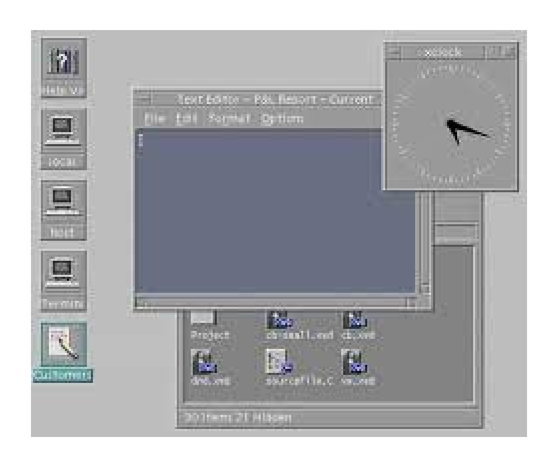
Motif es utilizado como infraestructura del ambiente de escritorio CDE.

Muchos desarrolladores argumentan que se ha vuelto obsoleto en comparación con otras bibliotecas como GTK o Qt, pero continúa siendo utilizado por varios sistemas antiguos.

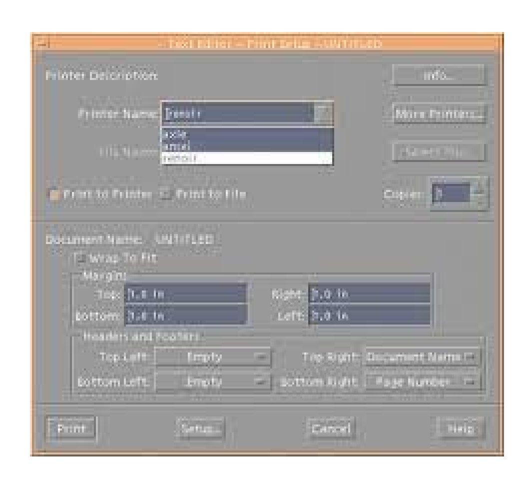
Existe una implementación oficial que tiene disponible el código fuente, llamada **OpenMotif** aunque partidarios del software libre consideran su licencia muy restrictiva. En 2012 OpenMotif se publica bajo licencia LGPL resolviendo la polémica.[1]

Una implementación no oficial con licencia GPL es **LessTif**.


plan
Adjust Time
Mon
Tue
Wed
Thu
Fri
Sat
Sun
Daylight Saving is
Cancel
Help
Done


xclock
File
Edit
Format
Options
Project
sourcefile.C


Z-Mail
Folder
Message
View
Find
Sort
Compose
Options
Layout
Help
New Arrivals
Read
Delete
Undelete
Save
Compose
Reply
Resend
Update
Forward
Command:


Text Editor - Print Setup - UNTITLED
Printer Description
Info...
Printer Name:
More Printers...
Print to Printer
Print to File
Copies:
Document Name: UNTITLED
Wrap To Fit
Margins
Top:
Right:
Bottom:
Left:
Headers and Footers
Top Left:
Top Right:
Bottom Left:
Bottom Right:
Empty
Document Name
Page Number
Print
Setup...
Cancel
Help

Para instalar Motif usaremos:

apt install libmotif-dev

Para compilar usaremos:

```
gcc -o programa programa.c -lXm -lXt -lX11
```

Para crear una ventana y un botón con un evento click:

```
#include <stdio.h>
#include <Xm/Xm.h>
#include <Xm/MainW.h>
#include <Xm/PushB.h>

void button_callback(Widget widget, XtPointer client_data, XtPointer call_data)
{
    printf("¡Hola! Has presionado el botón.\n");
}

int main(int argc, char **argv)
{
    XtAppContext app;
    Widget topLevel, mainWindow, button;

    topLevel = XtVaAppInitialize(&app, "VentanaEjemplo", NULL, 0, &argc, argv, NULL, NULL);
    mainWindow = XtVaCreateManagedWidget("mainWindow", xmMainWindowWidgetClass, topLevel, NULL);

    button = XtVaCreateManagedWidget("button", xmPushButtonWidgetClass, mainWindow, NULL);
    XtAddCallback(button, XmNactivateCallback, button_callback, NULL);

    XtRealizeWidget(topLevel);
    XtAppMainLoop(app);

    return 0;
}
```

Mostrar salida de un comando en una etiqueta:

```
#include <Xm/Xm.h>

#include <Xm/Text.h>
#include <stdio.h>
#include <stdlib.h>

int main(int argc, char **argv) {
    Widget topLevel, text;
    XtAppContext app;
    FILE *fp;
    char output[128];

    topLevel = XtVaAppInitialize(&app, "MotifExample", NULL, 0, &argc, argv, NULL, NULL);
    text = XtVaCreateManagedWidget("text", xmTextWidgetClass, topLevel, XmNeditable, False, NULL);

    fp = popen("ls", "r");
    if (fp == NULL) {
        perror("Error executing ls command");
        exit(1);
    }

    while (fgets(output, sizeof(output), fp) != NULL) {
        XmTextInsert(text, XmTextGetLastPosition(text), output);
    }
```

```
    pclose(fp);

    XtRealizeWidget(topLevel);
    XtAppMainLoop(app);

    return 0;
}
```